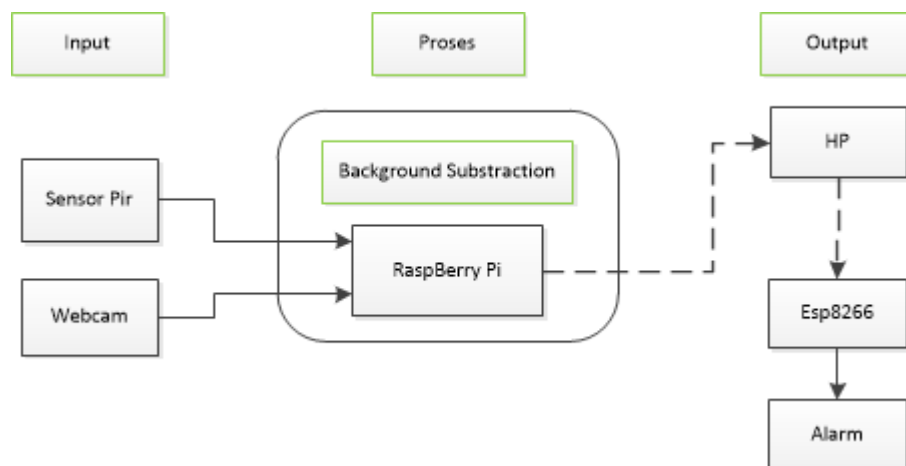


BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1. Analisis Sistem

Perangkat sistem keamanan perumahan, secara keseluruhan dari sistem yang akan dirancang menggunakan Blok diagram, Blok diagram merupakan salah satu bagian terpenting dalam perencanaan suatu alat, dari blok diagram inilah dapat diketahui cara kerja dari rangkaian keseluruhan yang digunakan. Sehingga keseluruhan blok diagram rangkaian tersebut akan menghasilkan suatu sistem yang dapat difungsikan atau dapat bekerja sesuai dengan perancangan. Berikut adalah blok diagram keseluruhan alat keamanan perumahan menggunakan metode *background subtraction* dan sensor PIR.



Gambar 3.1 Blok Diagram Sistem Pendeteksi

Keterangan diagram blok sebagai berikut :

a. *Webcam*

Webcam pada alat sistem keamanan ini akan digunakan untuk menangkap gambar dari suatu objek untuk mengamati suatu pergerakan yang akan terjadi.

b. Sensor Pir (*Passive InfaRed*)

Sensor Pir merupakan module atau alat yang akan digunakan untuk menangkap temperature suhu opada suatu objek yang bergerak yang akan menentukan objek manusia atau bukan.

c. *Raspberry Pi*

Raspberry Pi merupakan otak atau sistem control yang digunakan untuk memproses suatu citra dan penangkapan pada sensor pir. *Raspberry Pi* ini juga digunakan untuk tempat pemasangan *Webcam* dan Sensor Pir.

d. Background Substraction

Background Substraction adalah metode yang akan digunakan untuk menentukan suatu objek yang bergerak ada penguraian citra yang akan di proses seperti *greyscale* dan *threshold, filtering, frame*.

e. HP (*handphone*)

Hp pada sistem ini digunakan untuk melihat hasil foto objek yang sudah di tangkap oleh kamera yang sudah di proses oleh *Raspberry Pi*. Hasil foto tersebut bisa di lihat melalui *handphone* pemilik.

f. Esp8826

Esp8826 modul WiFi alat ini digunakan untuk menerima perintah yang akan di kirim oleh pemilik rumah setelah menerima hasil foto dari objek yang di tangkap oleh kamera Esp8826 ini di pasang berserta dengan Alarm.

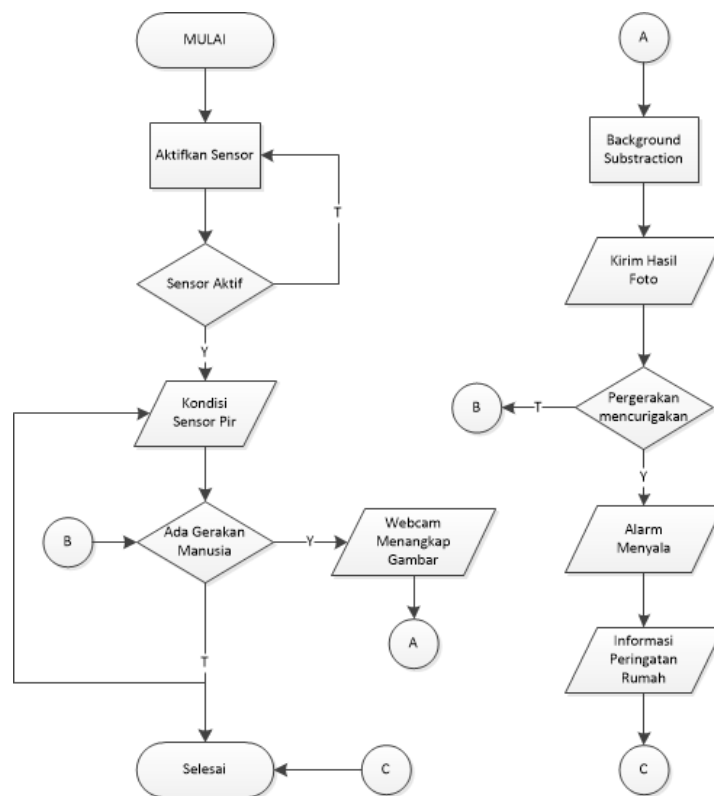
g. Alarm

Alarm perangkat ini di akan di pasangkan di tempat pos keamanan perumahan, alarm akan berbunyi jika menerima perintah dari pemilik rumah jika ada tindakan mencurigakan.

3.2. Hasil Analisis

3.2.1. Flowchart Sistem

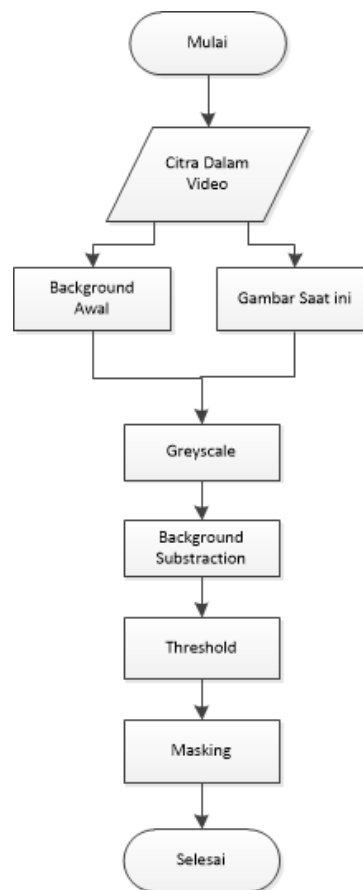
Flowchart adalah bentuk alir dari diagram blok yang merupakan salah satu bagian penting dalam perancangan suatu alat. Cara kerja keseluruhan alat yang akan dibuat dapat pula dilihat flowchart yang akan menghasilkan suatu sistem yang dapat difungsikan atau dapat bekerja.



Gambar 3.2 Flowchart Keseluruhan Sistem

3.2.2. Flowchart Pemrosesan Citra

Flowchart dibawah ini akan menjelaskan tentang cara kerja pemrosesan citra yang nantinya akan di ekstraksi, dibawah ini flowchart yang akan menghasilkan suatu sistem yang dapat difungsikan atau dapat bekerja.



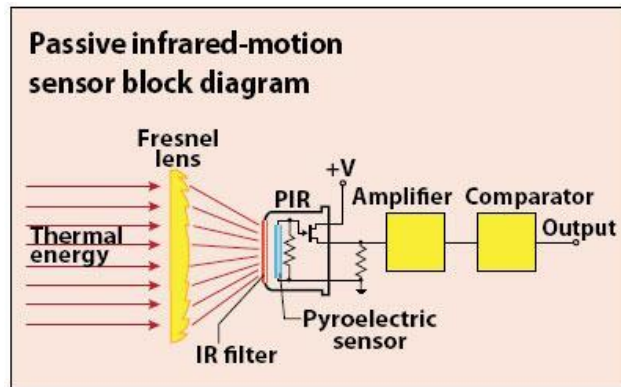
Gambar 3.3 Flowchart Pemrosesan Citra

3.3. Representasi Model

3.3.1. Pendeteksian Pada Sensor PIR

Pergerakan yang akan kita ambil atau kita proses disini adalah pergerakan objek manusia, Sesuai dengan namanya '*Passive*', sensor ini hanya merespon energi dari pancaran sinar *inframerah* pasif yang dimiliki oleh setiap benda yang terdeteksi. Prinsip atau cara kerja dari PIR ini adalah dengan mengukur energi *infra Red* yang di tangkap oleh sensor., maka *infra merah* yang diukur tersebut tidak dapat dihasilkan sendiri oleh komponen tersebut.

IR Filter dimodul sensor PIR ini mampu menyaring panjang gelombang sinar *inframerah* pasif antara 8 sampai 14 mikrometer, sehingga panjang gelombang yang dihasilkan dari tubuh manusia yang berkisar antara 9 sampai 10 mikrometer ini saja yang dapat dideteksi oleh sensor



Gambar 3.4 Komponen Sensor PIR

Seseorang akan berjalan melewati sensor, sensor akan menangkap pancaran sinar inframerah pasif yang dipancarkan oleh tubuh manusia yang memiliki suhu yang berbeda dari lingkungan sehingga menyebabkan material pyroelectric bereaksi menghasilkan arus listrik karena adanya energi panas yang dibawa oleh sinar inframerah pasif tersebut. Kemudian sebuah sirkuit ampliflier yang ada menguatkan arus tersebut yang kemudian dibandingkan oleh comparator sehingga menghasilkan output.

Untuk jarak jangkauan dari sensor PIR sendiri bisa disetting sesuai kebutuhan, akan tetapi jarak maksimalnya hanya +/- 10 meter dan minimal +/- 30 cm.

3.3.2. Tahap Pemrosesan Citra

Tahapan - tahapan pemrosesan citra yang akan diterapkan dalam aplikasi pendeteksi gerak ini masih menggunakan penangkapan kamera manual, belum secara otomatis dari Sensor PIR, bertujuan penangkapan kamera manual untuk menguji hasil pemrosesan metode *background subtraction* :

1. *Background registration*

Background registration merupakan tahap persiapan citra *background* yang akan digunakan pada proses *background subtraction*. Nilai citra *background* ini nantinya akan dikurangkan dengan citra video dimana objek manusia akan dideteksi. *background* dari hasil capture

rekaman pendeteksi gerakan yang berada disalah satu rumah. Terdapat 2 citra *background* yang digunakan adalah sebagai berikut :



Gambar 3.5 Backgroud Awal

Pada citra hasil penangkapan pada kamera didapat citra berikut :



Gambar 3.6 Citra Dari Video

2. Convert Grayscale

Citra *background* dan citra yang didapat dari *frame* video hasil rekaman memiliki *format* gambar RGB. Pada tahap ini setiap *frame* akan dirubah

menjadi *grayscale* agar lebih mudah diproses pada tahap selanjutnya. *Grayscale* merupakan warna-warna piksel yang berada dalam rentang gradasi warna hitam dan putih. Untuk merubah gambar RGB menjadi *grayscale* dapat dilakukan dengan persamaan (8) seperti berikut :

$$Y(x,y) = (0,229 * R) + (0,587 * G) + (0,114 * B)$$

Dimana :

Y = derajat keabuan

R = nilai *pixel* channel *Red*

G = nilai *pixel* channel *Green*

B = nilai *pixel* channel *Blue*

Pada citra *background* diambil sample acak citra ukuran 10x10 *pixel* pada suatu titik, citra *background* 10x10 adalah sebagai berikut :

R:127	R:126	R:125	R:125	R:124	R:123	R:123	R:123	R:126	R:129
G:120	G:117	G:116	G:116	G:115	G:114	G:114	G:114	G:117	G:122
B:110	B:108	B:107	B:106	B:106	B:105	B:105	B:105	B:108	B:112
R:128	R:128	R:125	R:122	R:122	R:122	R:126	R:126	R:127	R:129
G:121	G:121	G:11	G:113	G:113	G:113	G:119	G:119	G:120	G:122
B:111	B:111	B:108	B:104	B:104	B:104	B:109	B:109	B:110	B:112
R:131	R:131	R:128	R:124	R:125	R:126	R:129	R:129	R:129	R:129
G:124	G:124	G:121	G:115	G:116	G:117	G:122	G:122	G:122	G:122
B:114	B:114	B:111	B:106	B:107	B:108	B:112	B:112	B:112	B:112
R:129	R:131	R:130	R:128	R:127	R:127	R:129	R:129	R:130	R:129
G:125	G:124	G:123	G:121	G:120	G:120	G:122	G:122	G:123	G:125
B:114	B:114	B:113	B:111	B:110	B:110	B:112	B:112	B:113	B:114
R:130	R:129	R:129	R:131	R:129	R:128	R:131	R:131	R:133	R:133
G:123	G:122	G:122	G:124	G:122	G:121	G:124	G:124	G:126	G:129
B:115	B:112	B:114	B:114	B:112	B:111	B:114	B:114	B:116	B:118
R:131	R:129	R:131	R:131	R:133	R:134	R:134	R:134	R:134	R:133
G:124	G:122	G:12	G:124	G:126	G:127	G:127	G:127	G:127	G:126
B:118	B:114	B:116	B:116	B:118	B:119	B:117	B:117	B:117	B:116
R:131	R:130	R:132	R:133	R:134	R:135	R:133	R:133	R:133	R:133
G:124	G:123	G:125	G:126	G:127	G:12	G:126	G:126	G:126	G:126
B:118	B:115	B:117	B:118	B:119	B:120	B:116	B:116	B:116	B:116
R:131	R:131	R:133	R:134	R:135	R:135	R:130	R:130	R:131	R:132
G:124	G:124	G:126	G:127	G:128	G:128	G:123	G:123	G:124	G:125
B:116	B:116	B:116	B:117	B:118	B:118	B:113	B:113	B:114	B:115
R:130	R:131	R:133	R:134	R:134	R:134	R:129	R:129	R:129	R:131
G:123	G:124	G:126	G:127	G:127	G:127	G:122	G:122	G:122	G:123
B:113	B:116	B:116	B:117	B:117	B:117	B:112	B:112	B:112	B:114
R:129	R:130	R:132	R:132	R:132	R:132	R:128	R:128	R:129	R:131
G:123	G:123	G:126	G:126	G:125	G:125	G:121	G:121	G:122	G:124
B:111	B:113	B:114	B:114	B:115	B:115	B:111	B:111	B:112	B:114

Gambar 3.7 Nilai *Pixel* Citra Backgroud RGB

Untuk merubah menjadi *grayscale* nilai tersebut dihitung menggunakan persamaan sebagai berikut :

$$Y(0,0) = (0,299*127)+(0,587*120)+(0,114*110) = 112,063 \sim 112$$

$$Y(0,1) = (0,299*126)+(0,587*117)+(0,114*108) = 109,845 \sim 110$$

$$Y(0,2) = (0,299*125)+(0,587*116)+(0,114*107) = 108,915 \sim 109$$

$$Y(0,3) = (0,299*125)+(0,587*116)+(0,114*106) = 108,801 \sim 109$$

$$Y(0,4) = (0,299*124)+(0,587*115)+(0,114*106) = 107,985 \sim 108$$

$$Y(0,5) = (0,299*123)+(0,587*114)+(0,114*105) = 107,055 \sim 107$$

$$Y(0,6) = (0,299*123)+(0,587*114)+(0,114*105) = 107,055 \sim 107$$

$$Y(0,7) = (0,299*123)+(0,587*114)+(0,114*105) = 107,055 \sim 107$$

$$Y(0,8) = (0,299*126)+(0,587*117)+(0,114*108) = 109,845 \sim 110$$

$$Y(0,9) = (0,299*129)+(0,587*122)+(0,114*112) = 113,923 \sim 114$$

Nilai $Y(x,y)$ akan menggantikan 3 nilai piksel RGB menjadi 1 nilai *grayscale* sehingga untuk *pixel* pada koordinat (0,0) dengan nilai R=127, G=120, B=110 digantikan dengan nilai *pixel* $Y(0,0)$ yaitu 112. Begitu pula untuk nilai piksel berikutnya hingga *pixel* terakhir $Y(9,9)$.

Dengan perhitungan yang sama untuk setiap nilai *pixel* dalam matriks maka akan menghasilkan matriks citra *grayscale* sebagai berikut :

112	110	109	109	108	107	107	107	110	114
113	113	107	106	106	106	111	111	112	114
116	116	113	108	109	107	114	114	114	114
116	115	113	111	112	112	114	114	115	116
115	114	114	116	114	113	116	116	118	120
116	115	117	118	119	118	118	118	118	118
116	116	118	119	120	120	115	115	116	117
115	112	118	119	119	119	114	114	114	115
114	115	117	117	117	117	117	113	114	116
115	113	114	115	119	119	112	113	115	118

Gambar 3.8 Nilai Piksel Citra *Background* Hasil *Grayscale*

Setelah perhitungan konversi dari RGB ke *grayscale*, maka didapatkan citra *background* hasil *grayscale* sebagai berikut :



Gambar 3.9 Citra *Background* Awal Hasil *Greyscale*

Pada citra hasil dari video *frame* ini diambil citra ukuran 10x10 *pixel* pada suatu titik yang sama dengan permisalan pada citra *background* :

R:106 G:109 B:90	R:112 G:115 B:94	R:117 G:122 B:100	R:117 G:122 B:100	R:121 G:129 B:108	R:99 G:108 B:89	R:107 G:116 B:97	R:125 G:132 B:114	R:93 G:101 B:80	R:95 G:103 B:32
R:110 G:113 B:92	R:116 G:120 B:97	R:115 G:120 B:97	R:116 G:121 B:99	R:124 G:132 B:109	R:99 G:108 B:87	R:9108 G:117 B:96	R:123 G:131 B:110	R:93 G:101 B:80	R:98 G:103 B:83
R:111 G:114 B:93	R:117 G:121 B:98	R:113 G:119 B:93	R:116 G:121 B:9	R:125 G:133 B:110	R:96 G:105 B:84	R:109 G:118 B:97	R:121 G:129 B:108	R:91 G:99 B:78	R:98 G:103 B:81
R:108 G:111 B:88	R:114 G:118 B:93	R:112 G:118 B:92	R:115 G:120 B:97	R:122 G:130 B:106	R:91 G:101 B:77	R:110 G:120 B:96	R:119 G:127 B:104	R:85 G:93 B:70	R:98 G:103 B:81
R:105 G:109 B:86	R:111 G:115 B:90	R:113 G:117 B:92	R:116 G:121 B:98	R:119 G:127 B:103	R:87 G:95 B:72	R:112 G:120 B:97	R:119 G:127 B:104	R:80 G:88 B:65	R:97 G:102 B:78
R:105 G:109 B:86	R:112 G:116 B:93	R:110 G:114 B:91	R:120 G:124 B:101	R:120 G:125 B:103	R:89 G:94 B:72	R:115 G:120 B:98	R:117 G:122 B:100	R:79 G:84 B:62	R:196 G:101 B:78
R:106 G:110 B:87	R:114 G:118 B:95	R:114 G:116 B:94	R:119 G:122 B:101	R:119 G:122 B:101	R:89 G:94 B:72	R:117 G:122 B:102	R:116 G:121 B:101	R:74 G:79 B:59	R:101 G:106 B:84
R:107 G:111 B:85	R:116 G:120 B:95	R:113 G:117 B:92	R:117 G:121 B:98	R:116 G:119 B:98	R:91 G:94 B:75	R:121 G:124 B:107	R:117 G:121 B:106	R:74 G:78 B:63	R:108 G:110 B:96
R:107 G:111 B:86	R:116 G:120 B:95	R:111 G:115 B:90	R:114 G:118 B:95	R:114 G:117 B:98	R:91 G:94 B:77	R:119 G:121 B:107	R:115 G:118 B:107	R:76 G:79 B:68	R:86 G:88 B:77
R:117 G:111 B:85	R:114 G:118 B:91	R:116 G:120 B:93	R:121 G:125 B:102	R:114 G:117 B:98	R:85 G:87 B:73	R:98 G:100 B:89	R:93 G:94 B:88	R:64 G:65 B:60	R:46 G:45 B:41

Gambar 3.10 Nilai *Pixel* Citra *Frame* Video RGB

Untuk merubah menjadi *grayscale* nilai-nilai tersebut dihitung menggunakan persamaan sebagai berikut :

$$Y(0,0) = (0,299*106)+(0,587*109)+(0,114*90) = 98,157 \sim 99$$

$$Y(0,1) = (0,299*112)+(0,587*115)+(0,114*94) = 103,869 \sim 104$$

$$Y(0,2) = (0,299*117)+(0,587*122)+(0,114*100) = 109,807 \sim 110$$

$$Y(0,3) = (0,299*117)+(0,587*122)+(0,114*100) = 109,807 \sim 110$$

$$Y(0,4) = (0,299*121)+(0,587*129)+(0,114*108) = 115,744 \sim 116$$

$$Y(0,5) = (0,299*99)+(0,587*108)+(0,114*89) = 96,213 \sim 96$$

$$Y(0,6) = (0,299*107)+(0,587*116)+(0,114*97) = 103,653 \sim 104$$

$$Y(0,7) = (0,299*125)+(0,587*132)+(0,114*114) = 119,105 \sim 119$$

$$Y(0,8) = (0,299*93)+(0,587*101)+(0,114*80) = 89,704 \sim 90$$

$$Y(0,9) = (0,299*95)+(0,587*103)+(0,114*32) = 94,984 \sim 95$$

Nilai $Y(x,y)$ akan menggantikan 3 nilai piksel RGB menjadi 1 nilai *grayscale* sehingga untuk *pixel* pada koordinat (0,0) dengan nilai $R=106$, $G=109$, $B=90$ digantikan dengan nilai *pixel* $Y(0,0)$ yaitu 99. Begitu pula untuk nilai piksel berikutnya hingga *pixel* terakhir $Y(9,9)$.

99	104	110	110	116	96	104	119	90	95
102	108	108	109	118	96	104	118	90	92
103	109	106	108	119	93	105	116	88	92
100	106	105	108	116	89	107	114	82	92
98	103	105	109	114	84	107	114	77	91
98	104	102	112	113	84	108	110	74	92
99	106	105	110	110	84	104	109	70	97
99	108	105	108	107	85	112	109	70	95
99	108	104	107	106	85	113	110	70	98
102	106	108	113	106	79	91	87	60	42

Gambar 3.11 Nilai Piksel Citra *Frame* Video Hasil *Grayscale*

Setelah perhitungan konversi dari RGB ke *grayscale*, maka didapatkan citra *background* hasil *grayscale* sebagai berikut :



Gambar 3.12 Citra *Frame* Video Awal Hasil *Greyscale*

3. Background Subtraction

Setelah data *inputan* berupa video rekaman dan citra *background* melalui proses ekstraksi video *frames* dan *grayscale* maka akan didapatkan kumpulan video *frames* dan *background* image. Baik video *frames* atau *background* image telah melalui proses *grayscale* sehingga citra yang dihasilkan merupakan citra *grayscale*. Selanjutnya, citra video *frames* dan *background* akan dilakukan proses *background* subtraction.

Background subtraction atau Foreground Detection merupakan proses untuk mendeteksi pergerakan atau perbedaan yang signifikan dari video *frame*. *Frame* video akan dibandingkan dengan *background* images yang ada pada tahap sebelumnya. Proses ini akan mengurangi nilai matrix citra *current frame* dan nilai matrix citra *background* menggunakan perhitungan *absolut difference*.

Perhitungan *absolut difference* dapat dilakukan dengan persamaan (8) sebagai berikut :

$$Z(x,y) = |B(x,y) - C(x,y)|$$

dimana :

Z = nilai piksel absolute

B = nilai piksel *background*

C = nilai piksel *curent frame*

(x,y) = koordinat nilai piksel pada matriks

Nilai piksel hasil *grayscale* digunakan kembali pada proses ini. Nilai piksel *background* adalah sebagai berikut :

112	110	109	109	108	107	107	107	110	114
113	113	107	106	106	106	111	111	112	114
116	116	113	108	109	107	114	114	114	114
116	115	113	111	112	112	114	114	115	116
115	114	114	116	114	113	116	116	118	120
116	115	117	118	119	118	118	118	118	118
116	116	118	119	120	120	115	115	116	117
115	112	118	119	119	119	114	114	114	115
114	115	117	117	117	117	117	113	114	116
115	113	114	115	119	119	112	113	115	118

Gambar 3.13 Nilai Piksel Citra *Background* Hasil *Grayscale*

Sedangkan nilai piksel video *frame* yang akan digunakan adalah sebagai berikut :

99	104	110	110	116	96	104	119	90	95
102	108	108	109	118	96	104	118	90	92
103	109	106	108	119	93	105	116	88	92
100	106	105	108	116	89	107	114	82	92
98	103	105	109	114	84	107	114	77	91
98	104	102	112	113	84	108	110	74	92
99	106	105	110	110	84	104	109	70	97
99	108	105	108	107	85	112	109	70	95
99	108	104	107	106	85	113	110	70	98
102	106	108	113	106	79	91	87	60	42

Gambar 3.14 Nilai Piksel Citra *Frame* Video Hasil *Grayscale*

Setiap nilai piksel *background* dan video *frame* dihitung agar diketahui selisih nilai piksel *absolut* dari kedua citra. Kumpulan tersebut merupakan objek yang dapat disebut sebagai foreground mask. Perhitungannya adalah sebagai berikut :

$$Z(0,0) = | 112 - 99 | = 13$$

$$Z(0,1) = | 110 - 104 | = 6$$

$$Z(0,2) = | 109 - 110 | = 1$$

$$Z(0,3) = | 109 - 110 | = 1$$

$$Z(0,4) = | 108 - 116 | = 8$$

$$Z(0,5) = | 107 - 96 | = 11$$

$$Z(0,6) = | 107 - 104 | = 3$$

$$Z(0,7) = | 107 - 119 | = 12$$

$$Z(0,8) = | 110 - 90 | = 20$$

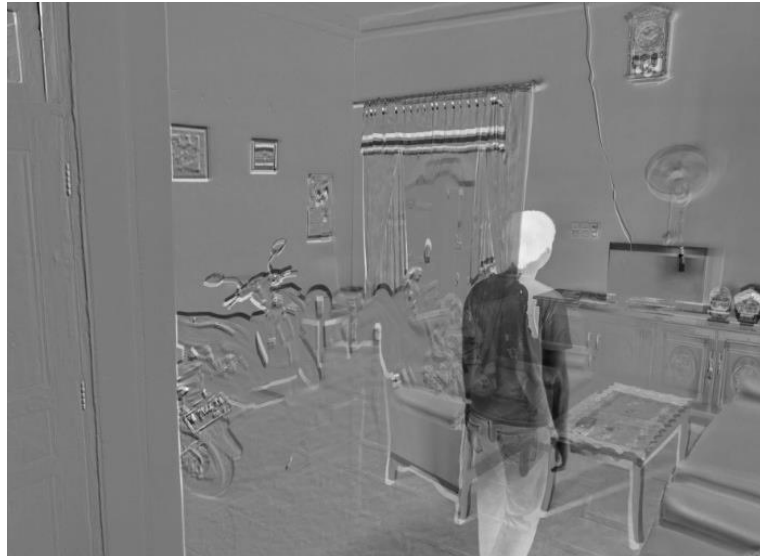
$$Z(0,9) = | 114 - 95 | = 19$$

Dari perhitungan tersebut diperoleh matriks hasil pengurangan atau matriks absolute difference sebagai berikut :

13	6	1	1	8	11	3	12	20	19
11	5	1	3	12	10	7	7	22	22
13	7	7	0	10	14	9	2	26	22
16	9	8	3	4	23	7	0	33	24
17	11	9	7	0	29	9	2	41	29
18	11	15	6	6	34	10	8	44	26
17	10	13	9	10	36	13	6	46	20
16	4	13	11	11	34	2	5	44	20
15	7	13	10	10	32	4	3	44	18
13	7	6	2	6	40	21	26	55	76

Gambar 3.15 Nilai Piksel Hasil absolut difference

Setelah perhitungan *background subtraction*, maka didapatkan citra hasil *background subtraction* sebagai berikut :



Gambar 3.16 Citra Hasil absolut difference

4. *Threshold*

Setelah melakukan *background subtraction* selanjutnya dilakukan *thresholding*. *Thresholding* merupakan teknik yang sederhana dan efektif untuk segmentasi citra. Proses *thresholding* sering disebut dengan proses binerisasi. Pada beberapa aplikasi pengolahan citra, terlebih dahulu dilakukan *threshold* terhadap citra gray level untuk dapat menjadi citra biner (citra yang memiliki nilai level keabuan 0 atau 255). Sebuah citra hasil proses *thresholding* dapat disajikan dalam histogram citra untuk mengetahui penyebaran nilai-nilai intensitas piksel pada suatu citra/bagian tertentu dalam citra sehingga untuk citra bimodal, histogram dapat dipartisi dengan baik (segmentasi objek dengan *background*) dan dapat ditentukan nilai *threshold*-nya.

Rumus untuk menentukan nilai *Threshold* bisa didapatkan dari persamaan sebagai berikut:

$$(x,y) = \{ \text{bernilai 1 jika } (x,y) \geq T \}$$

$$g(x,y) = \{ \text{bernilai } 0 \text{ jika } (x,y) < T \}$$

dimana :

$g(x,y)$ = nilai matrix citra hasil *thresholding*.

$f(x,y)$ = merupakan nilai matrix citra yang akan di-*threshold*.

T = merupakan nilai *threshold* (0 – 255)

Nilai *threshold* bergantung pada kebutuhan. Pada citra hasil *background subtraction* selanjutnya akan dilakukan *thresholding* dengan nilai $T=30$ sehingga proses perhitungannya adalah sebagai berikut :

13	6	1	1	8	11	3	12	20	19
11	5	1	3	12	10	7	7	22	22
13	7	7	0	10	14	9	2	26	22
16	9	8	3	4	23	7	0	33	24
17	11	9	7	0	29	9	2	41	29
18	11	15	6	6	34	10	8	44	26
17	10	13	9	10	36	13	6	46	20
16	4	13	11	11	34	2	5	44	20
15	7	13	10	10	32	4	3	44	18
13	7	6	2	6	40	21	26	55	76

Gambar 3.17 Nilai Piksel Hasil absolut difference

Untuk matriks (0,0)

$$f(0,0) = 6$$

$$f(0,0) = 13$$

$$g(0,0) = \{6 < 30\}$$

$$g(0,0) = \{13 < 30\}$$

$$g(0,0) = 0$$

$$g(0,0) = 0$$

Untuk matriks (0,2)

Untuk matriks (0,1)

$$f(0,0) = 1$$

$$g(0,0) = \{1 < 30\}$$

$$g(0,0) = 0$$

Untuk matriks (0,3)

$$f(0,0) = 1$$

$$g(0,0) = \{1 < 30\}$$

$$g(0,0) = 0$$

Untuk matriks (0,4)

$$f(0,0) = 8$$

$$g(0,0) = \{8 < 30\}$$

$$g(0,0) = 0$$

Untuk matriks (0,5)

$$f(0,0) = 11$$

$$g(0,0) = \{11 < 30\}$$

$$g(0,0) = 0$$

Untuk matriks (0,6)

$$f(0,0) = 3$$

$$g(0,0) = \{3 < 30\}$$

$$g(0,0) = 0$$

Untuk matriks (0,7)

$$f(0,0) = 12$$

$$g(0,0) = \{12 < 30\}$$

$$g(0,0) = 0$$

Untuk matriks (0,8)

$$f(0,0) = 20$$

$$g(0,0) = \{20 < 30\}$$

$$g(0,0) = 0$$

Untuk matriks (0,9)

$$f(0,0) = 19$$

$$g(0,0) = \{19 < 30\}$$

$$g(0,0) = 0$$

Dengan perhitungan yang sama untuk setiap nilai *pixel* dalam matriks dengan nilai $T=30$ maka akan menghasilkan matriks hasil *threshold* sebagai berikut :

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	1	0	0	1	0
0	0	0	0	0	1	0	0	1	0
0	0	0	0	0	1	0	0	1	0
0	0	0	0	0	1	0	0	1	0
0	0	0	0	0	1	0	0	1	1

Gambar 3.18 Nilai Citra *Pixel* Hasil *Threshold*

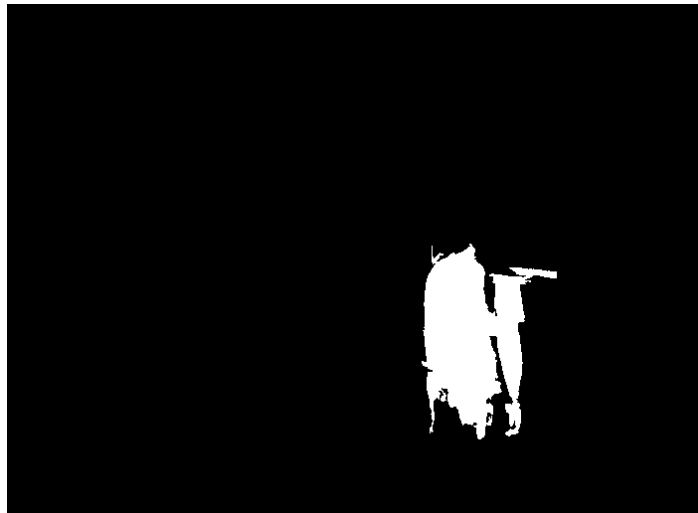
Setelah dilakukan *threshold* dengan nilai $T=30$, maka didapatkan citra hasil *threshold* sebagai berikut :



Gambar 3.19 Citra Hasil *Threshold*

Setelah dilakukan *threshold* seperti pada gambar diatas maka selanjutnya akan di lakukan fungsi penghapusan citra yang akan kita tentukan seblumnya jumlah ukuran *pixel* nya, disini kita akan menghapus semua objek yang lebih kecil dari 4000 piksel.

Setelah dilakukan penghapusan objek *pixel* yang kurang dari 4000, maka didapatkan citra hasil sebagai berikut :



Gambar 3.20 Citra Hasil Hapus Objek Kurang Dari 4000 *pixel*

5. *Masking* Citra

Fungsi *masking* digunakan untuk membuat ‘mask-image’. Citra mask adalah citra biner nilai piksel-piksennya 0 dan 1. Ketika citra mask digunakan dalam proses, maka piksel-piksel pada citra mask dengan nilai 1 diproses dan 0 diabaikan.

Maka nilai angka *pixel* 1 akan diberikan batas jalur (*masking*), setelah pemberian batas jalur kita akan melakukan cropping pada citra tersebut maka akan menghasilkan titik kordinat pada *pixel* yang akan kita gunakan. Setelah mendapatkan titik kordinat pada citra maka akan digabungkan pada citra pada video yang ditujukan pada gambar 3.5 dan hasil crop tersebut akan menghasilkan sebagai berikut :

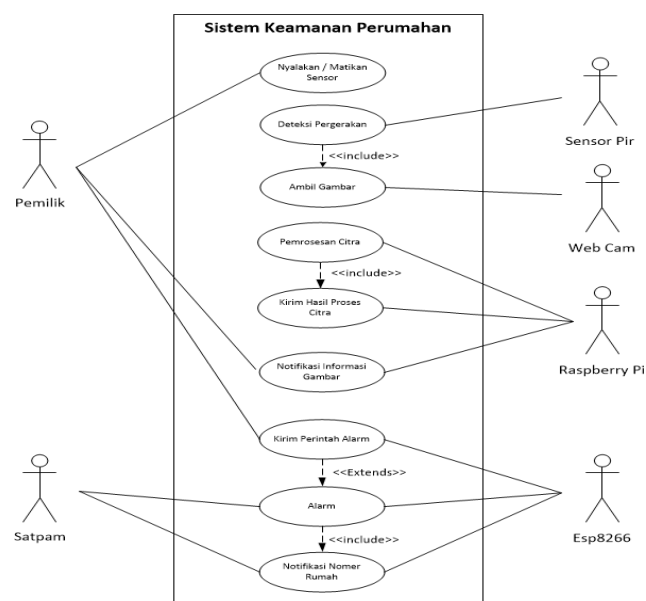


Gambar 3.21 Citra *Masking* Pada Video

3.4. Perancangan Sistem.

3.4.1. Use Case Diagram

Use Case menggambarkan hubungan antara entitas yang biasa disebut aktor dengan suatu proses yang dapat dilakukannya. Berikut merupakan gambar Use Case diagram dalam perancangan sistem keamanan rumah menggunakan sensor PIR.



Gambar 3.22 Use Case Diagram

Deskripsi pendefinisian aktor dan deskripsi pendefinisian use case pada sistem dapat dilihat pada tabel tabel berikut :

Tabel 3.1 Definisi Aktor

No.	Aktor	Deskripsi
1	Pemilik	Orang yang akan mengaktifkan sensor dan menerima notifikasi pergerakan pada rumah
2	Satpam	Orang yang akan menerima notifikasi keadaan rumah dan alarm
3	Sensor PIR	Modul perangkat yang akan mendeteksi pergerakan
4	<i>Raspberry Pi</i>	Perangkat <i>server</i> yang akan memproses dan mengirim notifikasi berupa citra atau gambar
5	Esp8266	menerima perintah dari pemilik dan membunyikan alarm
6	<i>Webcam</i>	Pengambilan citra atau gambar dari objek yang bergerak

Tabel 3.2 Definisi Use Case

No	Use Case	Deskripsi
1.	Aktifkan Sensor	Fungsionalitas untuk menyalakan sensor yang masih dalam keadaan mati
2.	Deteksi Pergerakan	Fungsionalitas untuk mendeteksi suatu pergerakan yang dilakukan oleh sensor
3.	Ambil gambar	Fungsionalitas untuk mengambil gambar keadaan rumah

4.	Pemrosesan Citra	Fungsionalitas untuk memproses citra kedalam <i>background</i> subtraction pada gambar yang di ambil
5.	Kirim hasil Proses Citra	Fungsionalitas untuk mengirim gambar yang sudah di <i>background</i> subtraction
6.	Notifikasi Informasi Gambar	Fungsionalitas untuk menerima data gambar dan informasi tentang keadaan rumah kepada pemilik
7.	Kirim Perintah Alarm	Fungsionalitas untuk mengirim perintah menyalakan alarm yang dilakukan oleh pemilik
8.	Alarm	Fungsionalitas untuk menginformasikan kepada petugas keamanan
9.	Notifikasi Nomer Rumah	Fungsionalitas untuk memberikan informasi rumah yang sedang dalam keadaan darurat

3.4.2. Skenario Use Case

Skenario use case mendeskripsikan langkah-langkah dalam proses sistem, baik dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor. Penjelasan skenario use case adalah sebagai berikut :

Tabel 3.3 Aktifasi Sensor

Identifikasi	
Nama	Aktifkan Sensor
Tujuan	Mengaktifkan Sensor PIR untuk mendeteksi pergerakan
Deskripsi	Pemilik rumah mengaktifkan sensor melalui smartphone

Aktor	pemilik
Skenario	
Kondisi Awal	Sensor dalam keadaan tidak aktif
Aksi Aktor	
Pemilik menyalakan sensor ketika sedang tidak dirumah	Sistem pada <i>Raspberry</i> akan menyalakan sensor PIR
Kondisi Akhir	Sensor aktif siap menangkap pergerakan

Tabel 3.4 Deteksi Pergerakan

Identifikasi	
Nama	Deteksi Pergerakan
Tujuan	Menangkap pergerakan manusia
Deskripsi	Sensor pir akan menangkap pergerakan pada manusia pada radius sensor
Aktor	Sensor PIR
Skenario	
Kondisi Awal	Sensor <i>standby</i> siap menangkap pergerakan
Aksi Aktor	
Sensor menangkap pergerakan	Meneruskan perintah ke <i>Raspberry Pi</i> yang nantinya mengambil gambar objek
Kondisi Akhir	Sensor mengirim perintah ke <i>Raspberry Pi</i>

Tabel 3.5 Ambil Gambar Citra

Identifikasi	
Nama	Ambil Gambar

Tujuan	Mengambil Gambar dari objek yang sedang bergerak terutama objek manusia
Deskripsi	Gambar yang di ambil dari citra yang nantinya akan di proses
Aktor	<i>Webcam</i>
Skenario	
Kondisi Awal	<i>Webcam</i> siap mengambil gambar dari objek
Aksi Aktor	
Mengambil Gambar Objek	<i>Webcam</i> akan mengambil objek dan akan diteruskan ke <i>Raspberry Pi</i>
Kondisi Akhir	Hasil penangkapan gambar atau citra

Tabel 3.6 Pemrosesan Citra

Identifikasi	
Nama	Pemrosesan Citra
Tujuan	Gambar yang di ambil akan di proses untuk menghasilkan marking pada objek yang bergerak agar gambar semakin jelas.
Deskripsi	Citra gambar di proses dalam Raspiberry Pi
Aktor	<i>Raspberry Pi</i>
Skenario	
Kondisi Awal	Gambar Hasil Captur Pada <i>Webcam</i>
Aksi Aktor	
Memproses dengan beberapa metode	Gambar akan mulai di lakukan proses dengan metode <i>background</i> subtraction
Kondisi Akhir	Hasil gambar yang sudah di proses menghasilkan <i>masking</i> pada objek yang bergerak

Tabel 3.7 Kirim Hasil Citra

Identifikasi	
Nama	Kirim Hasil Proses Citra
Tujuan	Gambar yang diproses akan di kirim ke pemilik rumah untuk dilakuka identifikasi
Deskripsi	Gambar yang dikirim berupa gambar yang sudah di proses <i>background subtraction</i> dan dilakukan <i>masking</i> .
Aktor	<i>Raspberry Pi</i>
Skenario	
Kondisi Awal	Gambar hasil pemrosesan
Aksi Aktor	
Mengirim gambar ke pemilik	Gambar akan diterima oleh pemilik rumah dan akan dilakukan identifikasi jika gambar mencurigakan.
Kondisi Akhir	Gambar dikiriimkan ke pemilik rumah

Tabel 3.8 Notifikasi Informasi

Identifikasi	
Nama	Notifikasi Informasi Gambar
Tujuan	Menginformasikan kepada pemilik jika ada pergerakan pada rumah berubah gambar
Deskripsi	Gambar atau citra yang diterima berupa informasi keadaan rumah pemilik
Aktor	<i>Raspberry Pi</i> , Pemilik
Skenario	
Kondisi Awal	Gambar diterima pemilik
Aksi Aktor	

Raspiberry Pi mengirim gambar ke Pemilik	Pemilik menerima gambar yang dikirimkan oleh <i>Raspberry Pi</i> prihan ada pergerakan manusia yang tertangkap oleh Sensor PIR
Kondisi Akhir	Hasil gambar berhasil terima oleh pemilik rumah.

Tabel 3.9 Kirim Perintah Alarm

Identifikasi	
Nama	Kirim Perintah Alarm
Tujuan	Mengaktifkan alarm yang terdapat di pos jaga perumahan
Deskripsi	Mengaktifkan alarm jika dirasa ada pergerakan mencurigakan pada rumah.
Aktor	Pemilik
Skenario	
Kondisi Awal	Mengaktifkan alarm
Aksi Aktor	
Pemilik Menekan tombol aktifkan alarm	Perintah aktifkan akan berfungsi dan nantinya alarm yang terdapat di pos jaga akan berbunyi
Kondisi Akhir	Perintah aktifkan alarm berhasil dikirim

Tabel 3.10 Alarm

Identifikasi	
Nama	Alarm
Tujuan	Akan Berbunyi ketika ada perintah aktifkan alarm

Deskripsi	Alarm akan di bunyikan ketika ada keadaan mencurigakan
Aktor	Esp8266
Skenario	
Kondisi Awal	Alarm Standby
Aksi Aktor	
Pemilik Menekan tombol aktifkan alarm	Alarm akan di aktifkan oleh modul esp8266 yang diaktifkan dari handphone pemilik dan alarm akan berbunyi
Kondisi Akhir	Alarm berbunyi

Tabel 3.11 Notifikasi Nomer Rumah

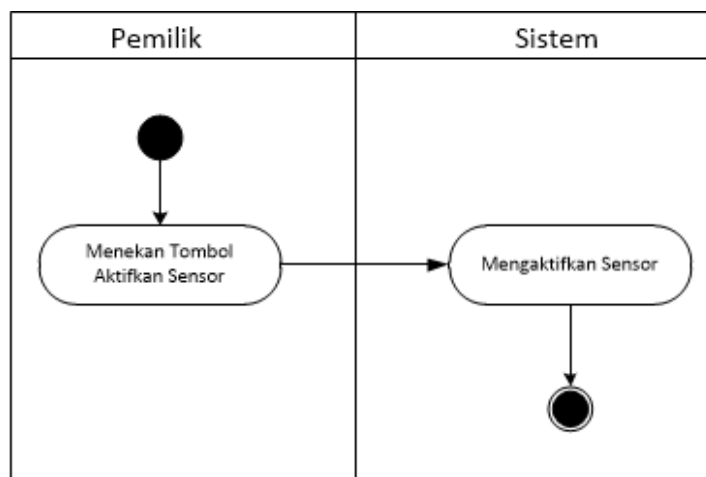
Identifikasi	
Nama	Notifikasi nomer rumah
Tujuan	Menginformasikan nomer rumah yang dalam keadaan darurat ketika alarm berbunyi
Deskripsi	Informasi nomer rumah kepada satpam atau penjaga perumahan
Aktor	Esp8266
Skenario	
Kondisi Awal	
Aksi Aktor	
Alarm berbunyi	Ketika alarm berbunyi otomatis modul Esp8226 akan menginformasikan nomer rumah pemilik yang sedang dalam keadaan darurat
Kondisi Akhir	Informasi nomer rumah pemilik

3.4.3. Activity Diagram

Activity diagram memodelkan aliran kerja dari urutan aktivitas dalam suatu proses yang mengacu pada use case diagram yang ada. Penjelasan masing-masing *Activity* diagram adalah sebagai berikut :

1. Activity Diagram Aktifkan Sensor

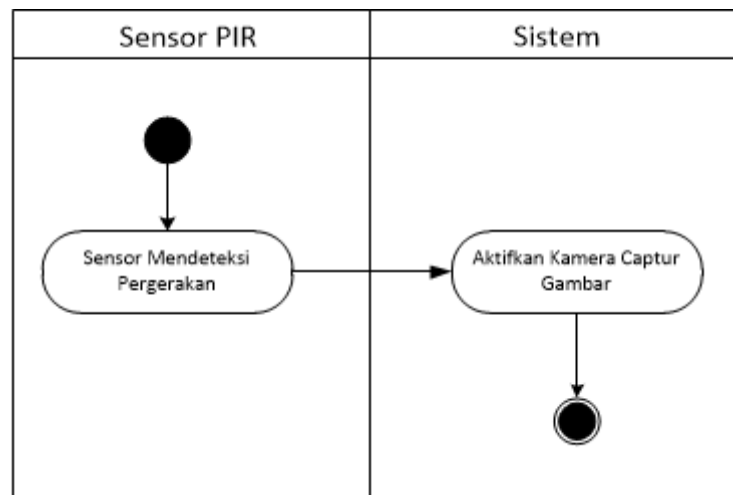
Activity Diagram Aktifkan Sensor menjelaskan aliran kerja pemilik rumah meaktifkan sensor, sebelum perangkat ini bekerja. Berikut ini *Activity* diagram aktifkan sensor :



Gambar 3.23 Activity Diagram Aktifkan Sensor

2. Activity Deteksi Pergerakan

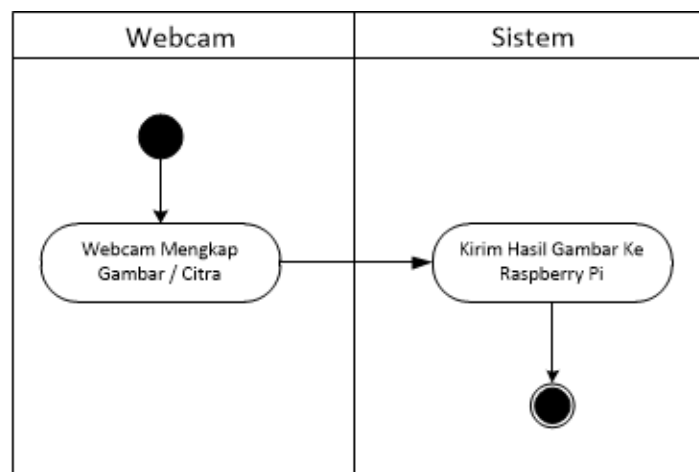
Activity Diagram Deteksi Pergerakan menjelaskan aliran kerja sensor PIR menangkap pergerakan pada kondisi rumah yang kosong. Berikut ini *Activity* diagram aktifkan sensor :



Gambar 3.24 Activity Diagram Deteksi Pergerakan

3. Ambil gambar / Citra

Activity Diagram Ambil Gambar menjelaskan aliran kerja *Webcam* menangkap gambar dari suatu pergerakan. Berikut ini Activity diagram ambil gambar :

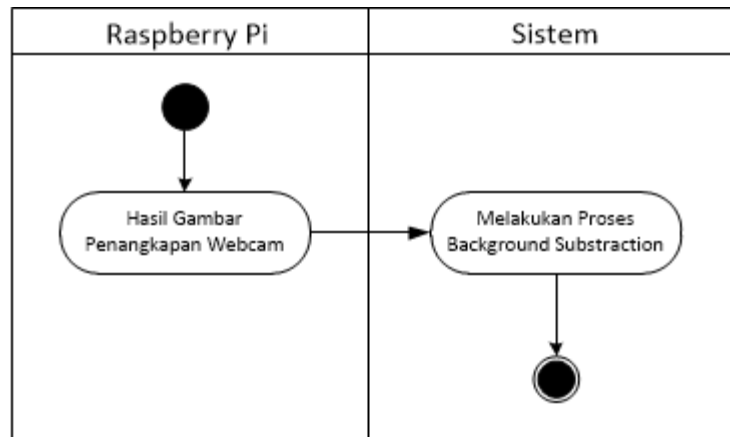


Gambar 3.25 Activity Diagram Ambil Gambar

4. Pemrosesan Gambar / Citra

Activity Diagram Pemrosesan Gambar menjelaskan aliran kerja *Raspberry Pi* memproses gambar dari *Webcam* dimulai dari gambar di ubah menjadi

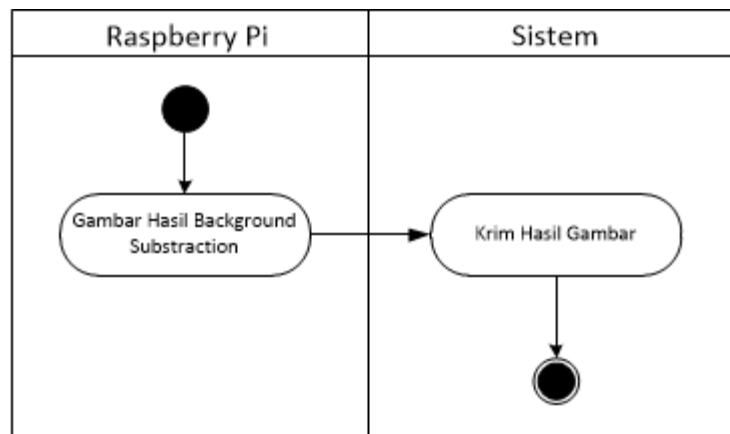
background subtraction, threshold dan masking. Berikut ini *Activity diagram* Pemrosesan gambar :



Gambar 3.26 Activity Diagram Pemrosesan Gambar

5. Kirim Hasil Proses Gambar

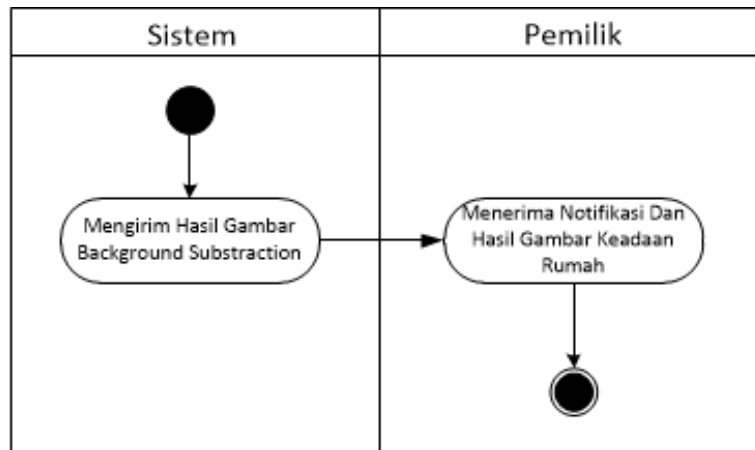
Activity Diagram Kirim Hasil Proses Citra menjelaskan aliran kerja *Raspberry Pi* mengirim gambar ke pemilik rumah setelah gambar atau citra selesai di proses. Berikut ini *Activity diagram* Kirim hasil proses citra:



Gambar 3.27 Activity Diagram Hasil Proses Gambar

6. Notifikasi Informasi Gambar

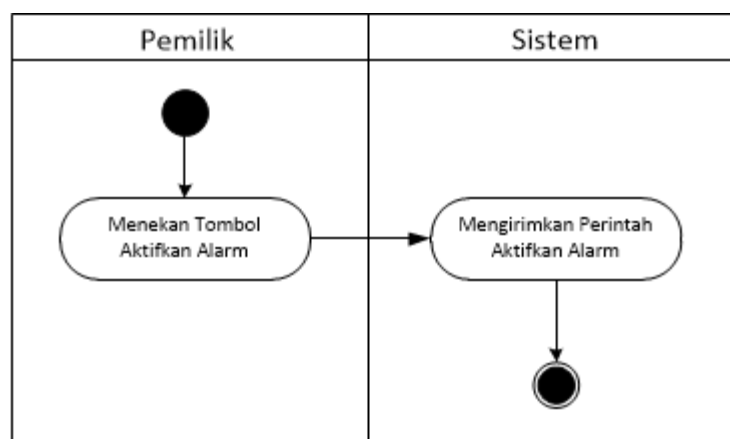
Activity Diagram Notifikasi Informasi Gambar menjelaskan aliran kerja Pemilik menerima hasil gambar yang sudah dalam bentuk citra hasil *background subtraction*. Berikut ini *Activity diagram* Hasil Notifikasi Informasi Gambar :



Gambar 3.28 *Activity Diagram* Notifikasi Gambar

7. *Activity* Kirim Perintah Alarm

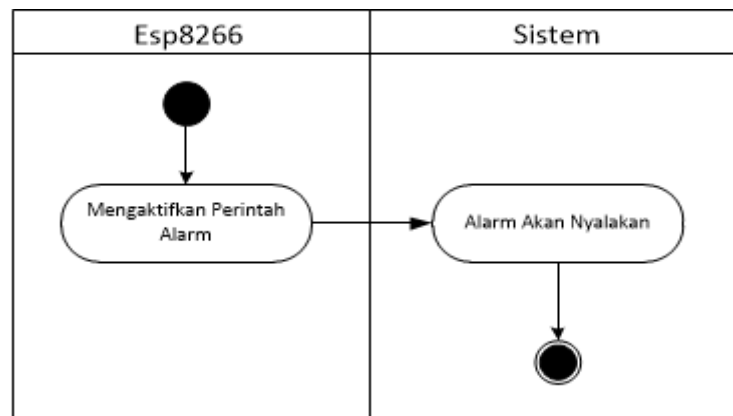
Activity Diagram Kirim Perintah Alarm menjelaskan aliran kerja Pemilik akan menekan tombol aktifkan alarm untuk menginformasikan kepada petugas keamanan. Berikut ini *Activity diagram* Hasil Kirim Perintah Alarm :



Gambar 3.29 *Activity* Kirim Perintah Alarm

8. Activity Alarm

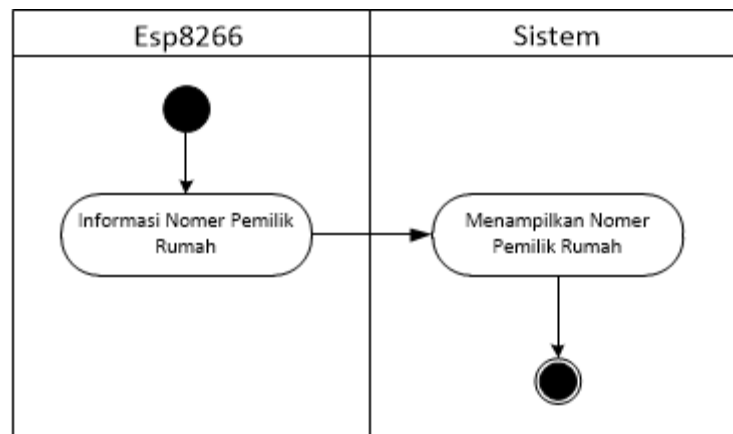
Activity Diagram Kirim Perintah Alarm menjelaskan aliran kerja Modul Esp8266 akan menggerakkan alarm agar bisa berbunyi untuk mnghinformasikan kepada penjaga keamanan. Berikut ini Activity diagram Alarm :



Gambar 3.30 Activity Alarm

9. Activity Notifikasi Nomer Rumah

Activity Diagram Notifikasi Nomer Rumah menjelaskan aliran kerja Modul Esp8266 akan mnghinformasikan kepada penjaga keamanan nomer rumah berapa yang dalam keadaan darurat. Berikut ini Activity diagram Alarm :



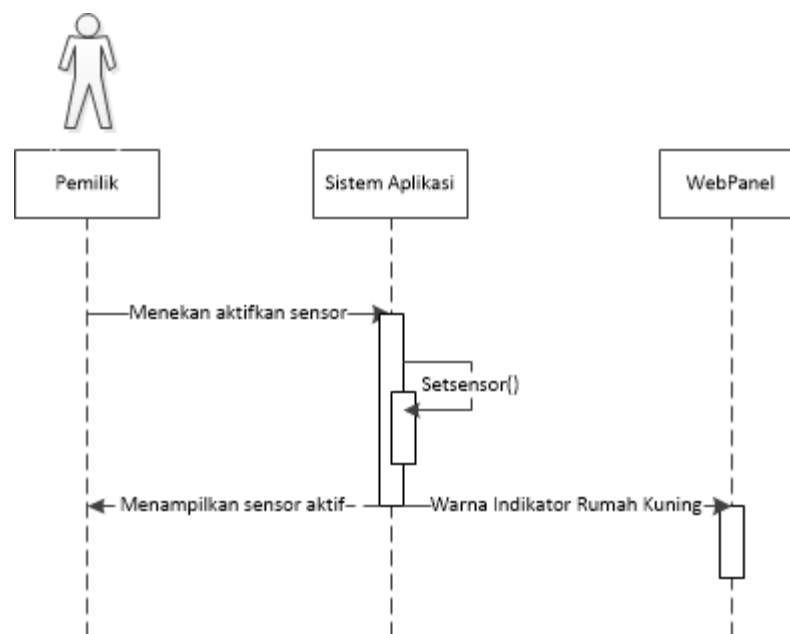
Gambar 3.31 Activity Notifikasi Nomer Rumah

3.4.4. Sequence Diagram

Sequence diagram adalah suatu diagram interaksi yang menekankan pada pengaturan waktu dari pesan-pesan. Diagram ini menampilkan sekumpulan peran dan pesan-pesan yang dikirim dan diterima oleh instansi yang memegang peranan tersebut. Sequence diagram menangkap objek dan class yang terlibat dalam skenario dan urutan pesan yang ditukar antara objek diperlukan untuk melaksanakan fungsionalitas skenario. Sequence diagram berasosiasi dengan use case selama proses pengembangan.

1. Sequence Diagram Aktifkan Sensor

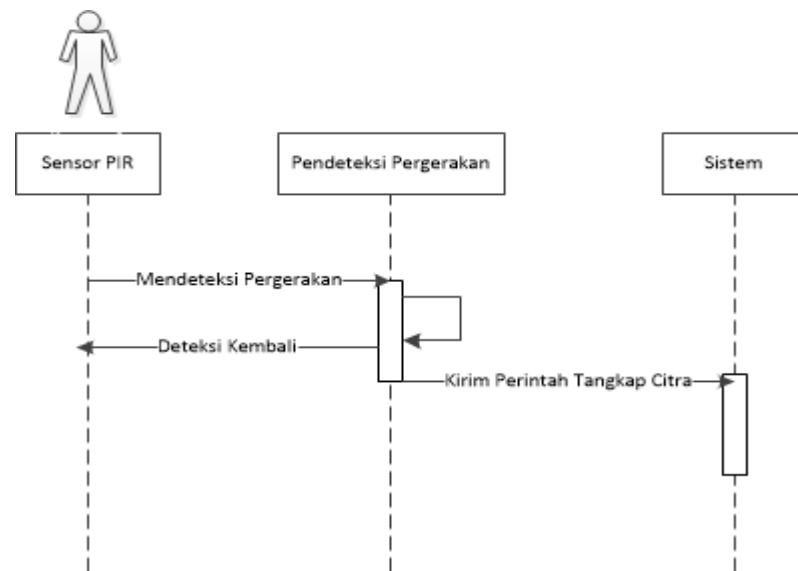
Sequence diagram Aktifkan Sensor menunjukkan rangkaian pesan yang dikirim antara objek pemilik dengan objek lainnya dalam melakukan proses aktifkan sensor ke dalam sistem, sequence diagram Aktifkan Sensor dapat dilihat sebagai berikut :



Gambar 3.32 Sequence Diagram Aktifkan Sensor

2. Sequence Diagram Deteksi Pergerakan

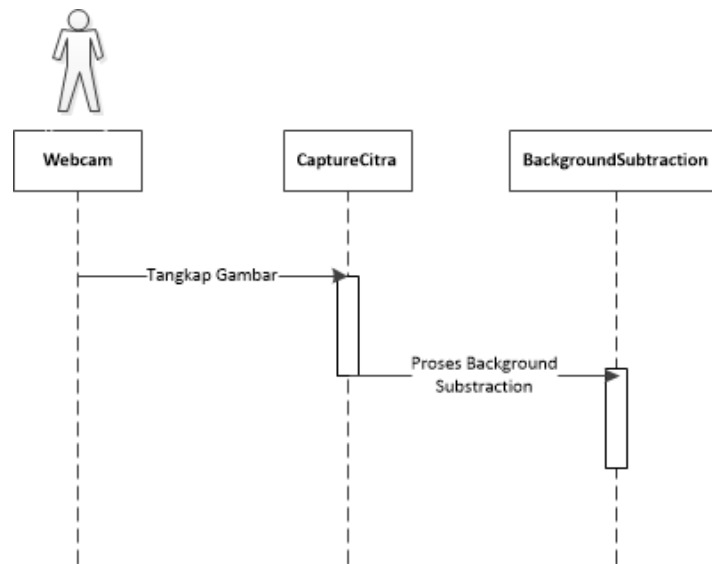
Sequence diagram deteksi pergerakan menunjukkan rangkaian pesan yang dikirim antara objek sensor dengan objek lainnya dalam melakukan proses pendeteksi pergerakan ke dalam sistem, sequence diagram deteksi pergerakan dapat dilihat sebagai berikut :



Gambar 3.33 Sequence Diagram Deteksi Pergerakan

3. Sequence Diagram Ambil Gambar

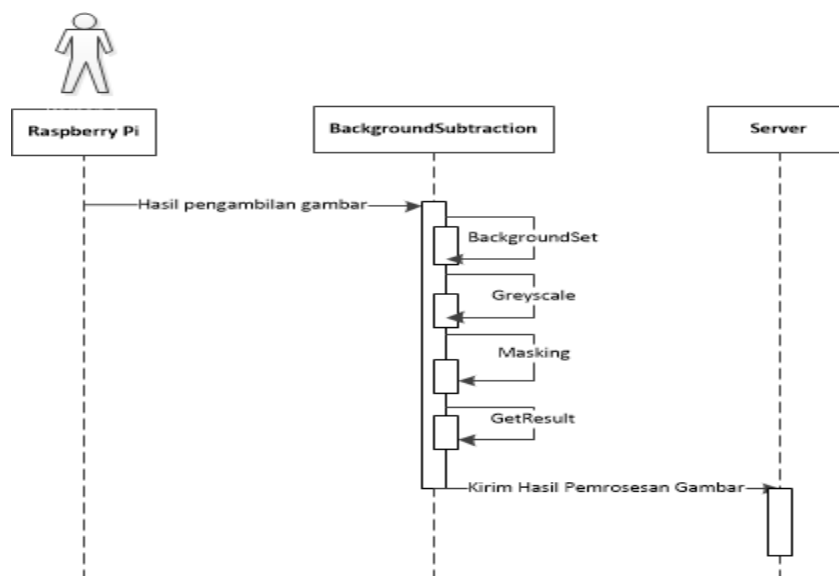
Sequence diagram ambil gambar menunjukkan rangkaian pesan yang dikirim antara objek *Webcam* dengan objek lainnya dalam melakukan proses penangkapan gambar ke dalam sistem, sequence diagram ambil citra atau gambar dapat dilihat sebagai berikut :



Gambar 3.34 Sequence Diagram Ambil Gambar

4. Sequence Diagram Pemrosesan Gambar

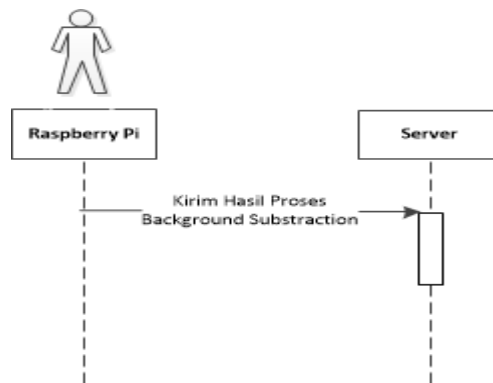
Sequence diagram ambil gambar menunjukkan rangkaian pesan yang dikirim antara objek *Raspberry Pi* dengan objek lainnya dalam melakukan pemrosesan gambar ke dalam sistem, sequence diagram pemrosesan citra atau gambar dapat dilihat sebagai berikut :



Gambar 3.35 Sequence Diagram Pemrosesan Gambar

5. Sequence Diagram Kirim Gambar

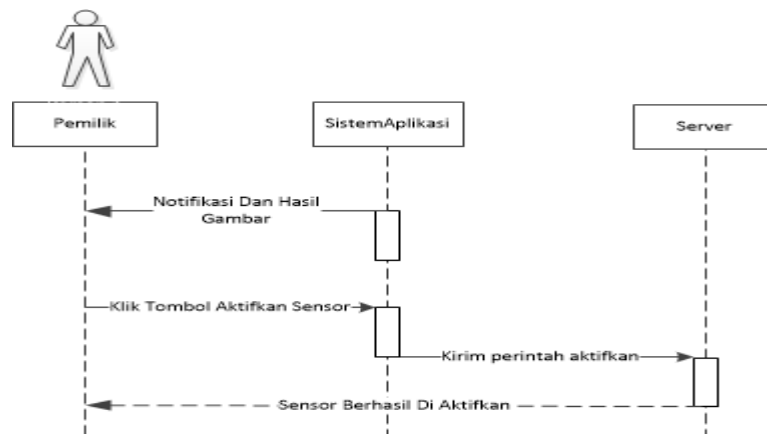
Sequence diagram Kirim Gambar menunjukkan rangkaian pesan yang dikirim antara objek *Raspberry Pi* dengan objek lainnya dalam melakukan kirim gambar ke dalam sistem, sequence diagram kirim citra atau gambar dapat dilihat sebagai berikut :



Gambar 3.36 Sequence Diagram Kirim Gambar

6. Sequence Diagram Notifikasi Dan Kirim Perintah Alarm

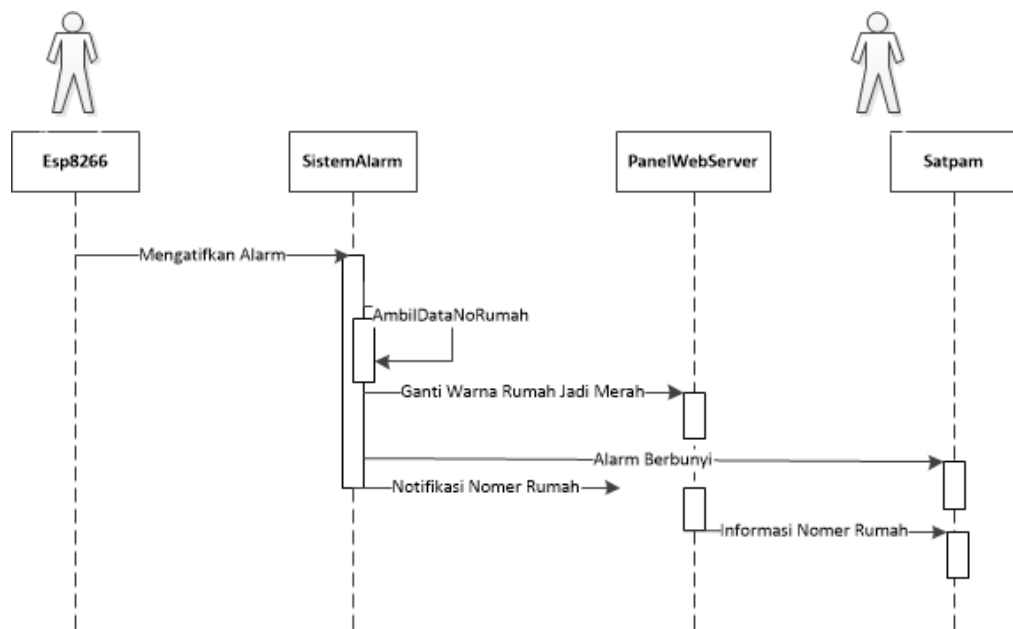
Sequence diagram notifikasi gambar menunjukkan rangkaian pesan yang dikirim antara objek *Server* Dari *Raspberry Pi* ke pemilik dan *server* akan mengirimkan perintah notifikasi ke dalam *server*, sequence diagram notifikasi dan kirim perintah alarm dapat dilihat sebagai berikut :



Gambar 3.37 Sequence Diagram Notifikasi Dan Kirim Perintah Alarm

7. Sequence Diagram Alarm Dan Notifikasi Nomer Rumah

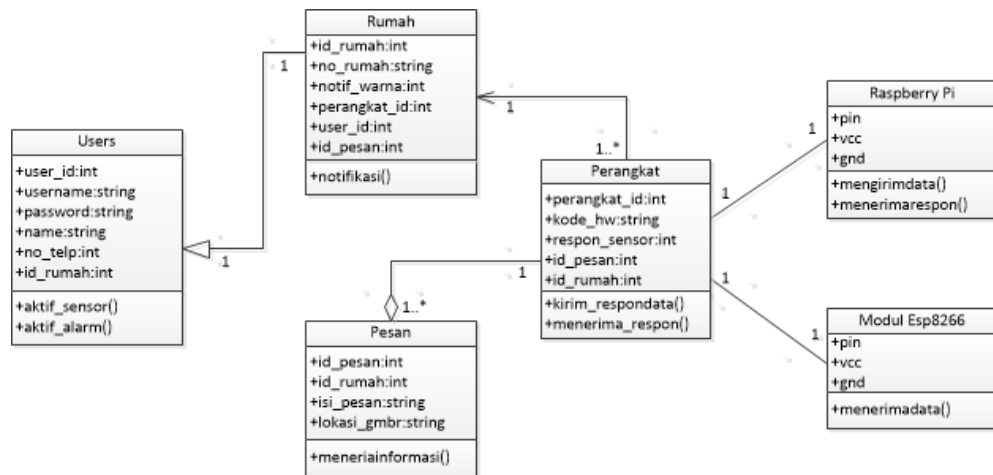
Sequence diagram alarm dan notifikasi nomer rumah menunjukkan rangkaian pesan yang dikirim antara objek esp8266 dari *server* ke alarm, kemudian akan di terima oleh keamanan, sequence diagram alarm dan notikasi nomer rumah dapat dilihat sebagai berikut :



Gambar 3.38 Sequence Diagram Alarm Dan Notifikasi Nomer Rumah

3.4.5. Class Diagram

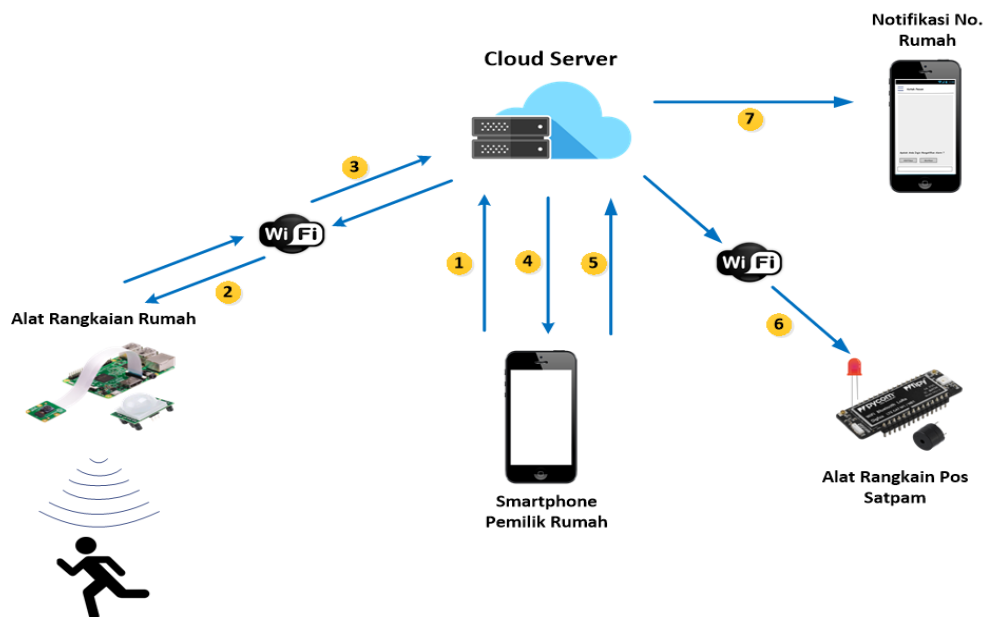
Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Struktur ini meliputi atribut-atribut dan metode-metode yang ada pada masing-masing kelas. Class diagram yang akan dibangun dapat dilihat pada gambar berikut :



Gambar 3.39 Desain Class Diagram

3.5. Perancangan Perangkat

Pada sub bab ini akan membahas tentang perancangan perangkat keras. Perancangan perangkat keras meliputi perancangan modul mikrokontroller *Raspberry Pi*, *Esp8266*, *Webcam*, Sensor PIR dan yang mendukung sistem kerja alat ini



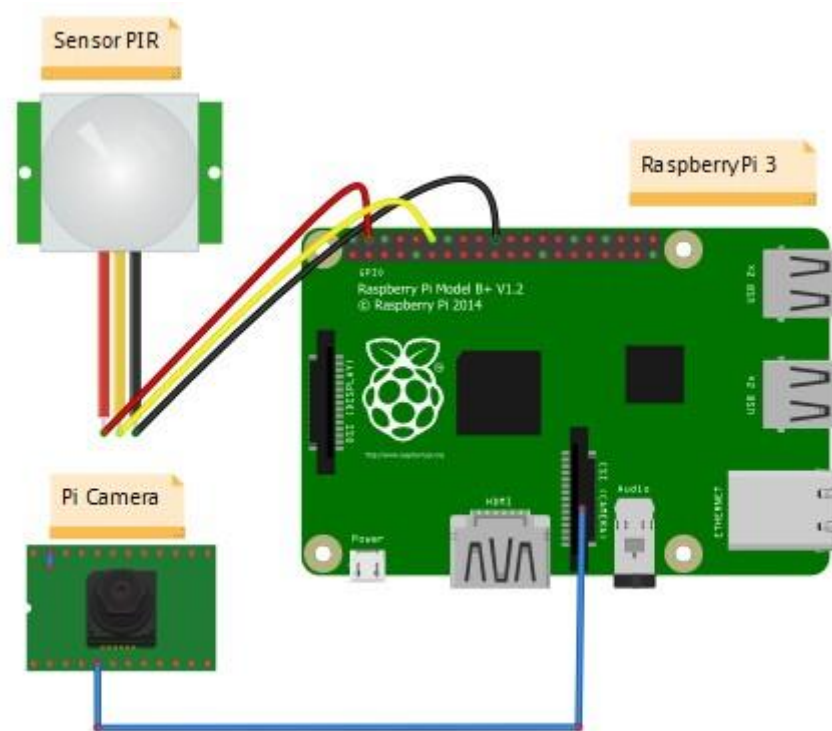
Gambar 3.40 Perancangan Keseluruhan Perangkat Sistem

Perancangan perangkat keras pada gambar diatas adalah keseluruhan sistem yang nantinya akan berjalan dimulai dari mengaktifkan sensor sampai alarm berbunyi :

1. Aktifkan sensor / perangkat.
2. Perangkat aktif untuk menangkap pergerakan manusia.
3. Mengirim hasil gambar yang sudah proses ekstraksi *background subtraction*.
4. Menerima hasil gambar yang sudah ekstraksi.
5. Mengirim perintah aktifkan alarm.
6. Alarm berbunyi.
7. Informasi nomer rumah ke petugas keamanan (satpam).

3.5.1. Perancangan Perangkat Pada Rumah

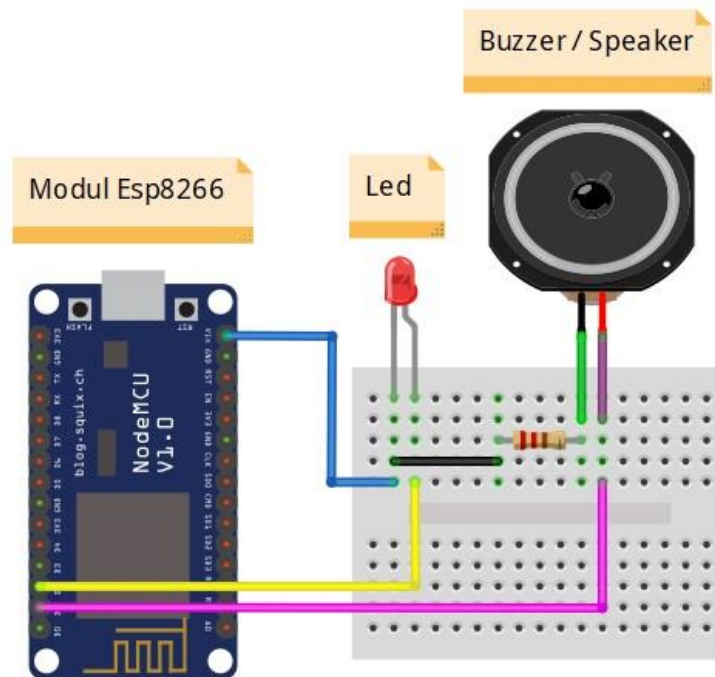
Perancangan perangkat keras yang akan di pasang pada setiap rumah, Sensor PIR akan di hubungkan ke *Raspberry Pi* Menggunakan jumper, setiap jumper mempunyai masing – masing pin yang akan di hubungkan, kamera akan di pasang di slot kamera yang sudah tersedia, nantinya *Raspberry Pi* akan menggunakan koneksi Wifi untuk terhubung ke internet, dibawah ini yang nantinya akan di rangkat di setiap rumah :



Gambar 3.41 Rangkain Perangkat Pada Rumah

3.5.2. Perancangan Perangkat Pada Pos Keamanan

Perancangan perangkat keras yang akan di pasang pada pos satpam, yaitu Modul Esp8266, Led dan *Buzzer*, semua rangkaian akan terhubung ke Esp8266, dan yang pasti harus selalu terhubung ke jaringan internet, dibawah ini yang nantinya akan di rangkat di setiap rumah :



Gambar 3.42 Rangkain Perangkat Pada Pos Satpam

3.6. Perancangan Antar Muka

Perancangan antarmuka merupakan sebuah penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi. Adapun perancangan antarmuka sistem yang akan dibuat adalah sebagai berikut :

3.6.1. Perancangan Antarmuka Pemilik Rumah

Perancang dibawah ini yang akan dilihat dan dipakai oleh pemilik rumah, dibawah ini antarmuka pemilik rumah :

1. Tampilan Aktifkan Sensor

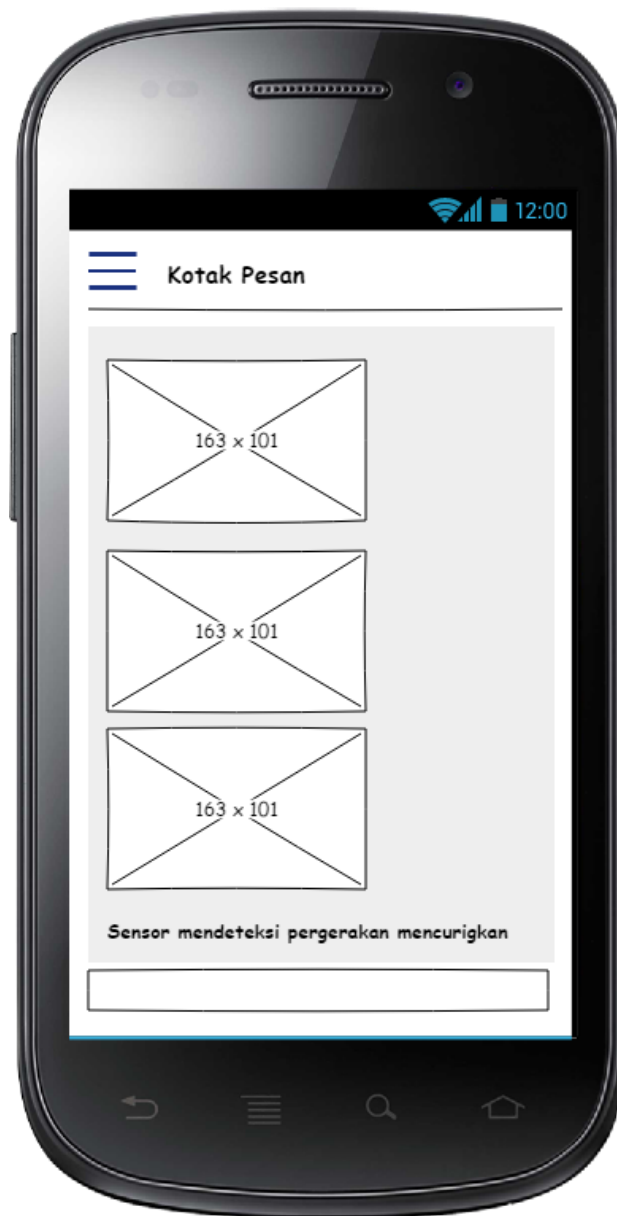
Dibawah ini adalah tampilan halaman utama pada pemilik rumah,halaman utama ini untuk mengaktifkan dan mematikan perangkat yang terdapat di rumah, dibawah ini tampilan antarmuka halaman utama pemilik :



Gambar 3.43 Antramuka Aktifkan Alarm

2. Tampilan Kotak Pesan

Dibawah ini adalah tampilan halaman utama pada pemilik rumah,halaman utama ini untuk mengaktifkan dan mematikan perangkat yang terdapat di rumah, dibawah ini tampilan antramuka halaman utama pemilik :



Gambar 3.44 Antramuka Halaman Notif Gambar

3.6.2. Perancangan Antarmuka Satpam

Perancang dibawah ini yang akan dilihat dan dipakai oleh satpam, dibawah ini antarmuka satpam:



Gambar 3.45 Antramuka Halaman Satpam

3.7. Pengujian Sistem

Setelah alat keamanan pada perumahan selesai dibuat, maka langkah selanjutnya adalah pengujian alat pendeteksi pergerakan dan sistem yang kita buat. Adapun pengujian yang dilakukan di antaranya :

1. Pengujian sensor PIR
2. Pengujian hasil proses ekstraksi citra
3. Pengujian respon alarm

3.7.1. Tahap Pengujian perangkat dan sistem

Pengujian perangkat dan sistem akan dilakukan dengan langkah – langkah dibawah ini :

1. Aktifkan Sensor

Sebelum perangkat bekerja, sensor harus mengaktifkan perangkat terlebih dahulu.

2. Pendeteksian Sensor

Ketika perangkat sudah diaktifkan maka sensor PIR, akan mendeteksi semua pergerakan pada manusia.

3. Penangkapan Citra Atau Gambar

Setelah sensor mendeteksi adanya pergerakan maka kamera akan menangkap gambar atau citra.

4. Pemrosesan Citra

Pemrosesan akan dilakukan setelah kamera mengambil gambar, *Raspberry Pi*, akan melakukan ekstraksi citra, setelah hasil ekstraksi citra selesai, akan *Raspberry Pi* akan mengirim ke smartphone pemilik rumah.

5. Aktifkan Alarm

Pemilik rumah akan menerima gambar hasil citra yang sudah di ekstraksi, setelah melihat hasil gambar, pemilik bisa menekan tombol aktifkan alarm, jika ada pergerakan mencurigakan. alarm akan berbunyi di sertai informasi nomer pemilik rumah.

3.7.2. Skenario Pengujian

Pada tahap pengujian terdiri dari beberapa skenario pengujian yang dilakukan dalam beberapa kondisi seperti berikut:

- a. Pengujian metode background subtraction akan dilakukan untuk mendeteksi objek bergerak
- b. Pengujian sensor akan di coba dari beberapa sudut dan maksimal jarak 5m dari sensor bertujuan apakah sensor mendeteksi adanya pergerakan dengan baik.
- c. Sensor akan diujikan untuk mendeteksi objek pada manusia
- d. Pengujian alat akan di hitung *delay* waktu proses awal menyalakan sensor sampai data pengiriman gambar dikirim kepemilik rumah dan proses menyalakan alarm sampai berbunyi.

Pengujian perangkat sistem diperlukan untuk memastikan apakah sistem yang telah dibuat sudah berjalan sesuai dengan apa yang diharapkan.